LISTING OF CLAIMS

1. (currently amended) A method for processing data packets of a data stream in a communication system, the method comprising:

depending on a predetermined feature of a data packet, said data packet being one of a plurality of data packets received in an original packet order, processing the data packet as one of a slow data packet on a slower path or a fast data packet on a faster path, wherein the data packet is processed faster in the faster path than in the slower path; and

reordering the data packets after the processing into the original packet order ~~they had prior to the processing~~;

wherein the reordering comprises the steps of:

for each fast processed fast data packet processed on the faster path, determining whether one or more slow data packet precede the fast data packet in the original packet order; determining whether said one or more slow data packets have been processed; and, if one or more slow data packets that preceded said fast data packet in said original packet order has not been processed, storing one or more fast processed ~~the~~ fast data packets that were processed on the faster path in a memory until said one or more ~~if not all the~~ slow data packets ~~that before the processing were in order before the fast data packets and were processed on the slower path are~~ have been processed and received at an output; and,

fetching the stored one or more fast processed fast data packets from the memory for ~~and~~ outputting to the output when all the slow data packets that preceded the one or more fast data packets in the original packet order have been ~~before the processing were in order before the fast data packets are~~ received at the output.

2. (currently amended) A method as recited in claim 1, comprising:

producing a sync signal if a last slow data packet is followed by a fast data packet;

giving the sync signal to the slower processing path after the last slow data packet;

generating a ready signal when the sync signal is processed in the slower path; and,

in response to the ready signal, taking ~~giving~~ the stored fast processed data packets out of the memory and delivering further fast processed data packets to the output directly after the stored fast data packets ~~are~~ have been drained out of the memory.

3. (currently amended) A method as recited in claim 1, comprising:

counting the number of consecutive slow data packets ~~that are in order~~ before a given fast data packet in the original packet order prior to ~~before the~~ processing;

storing ~~the~~ one or more fast processed fast data packets in the memory if not all counted consecutive slow data packets have been ~~are yet~~ processed;

~~giving~~ taking the stored one or more fast processed fast data packets out of the memory to the output, when the counted slow data packets have been processed and given out to the output; and,

assigning further fast processed data packets to the output directly after the stored fast processed fast data packets have been ~~are~~ drained out of the memory to the output.

4. (original) A method as recited in claim 1, comprising:

processing at least some data flows simultaneously;

processing the data flows independently; and,

processing slow and fast data packets of the same data flow in order within the data flow.

5. (currently amended) A method as recited in claim 1, comprising:

processing a first data packet of a data flow in the slow path to generate ~~generating~~ features from the data packet;

storing the generated features; and, processing ~~the following~~ successive data packets of the data flow in the fast path using the stored features.

6. (currently amended) A method as recited in claim 5, comprising:

determining the features by processing a header of the data packet;

determining from the features a direction to deliver to the data packets of the data flow;

storing the direction; and,

sending successive ~~following~~ data packets of the data flow based on the stored direction.

7. (currently amended) A method as recited in ~~any of~~ claim 1, comprising:

counting, via a counter, a number of ~~for~~ slow data packets that are delivered to an input queue of the slower path;

~~counting down~~ decrementing the number in ~~by~~ the counter by one for each ~~if a~~ processed slow data packet that leaves the slower path;

storing processed fast data packets that ~~are~~ have been processed by ~~given out of~~ the faster path in a memory if the number of the counter is higher than a predetermined value; and,

draining stored fast data packets out of the memory to the output ~~if~~ when the number of the counter equals the predetermined value; and, ~~giving~~ providing successive ~~further~~ processed fast data packets out directly to the output.

8. (currently amended) A data processing system comprising: an input connected to a first distributing unit selectively connectable to an input of a slower processing unit and an input of a faster processing unit having an output connected to an input of a second distributing unit selectively connectable to a system output and a memory, wherein an output of the memory and an output of the slower processing unit are connected to the system output, wherein:

the first distributing unit comprises at least one component for checking~~, in use, checks~~ a predetermined feature of a data packet, said data packet being one of a plurality of data packets received in an original packet order, and assigning ~~assigns~~ the data packet to the slower or faster processing unit in dependence on the feature of the data packet, ~~wherein~~

the second distributing unit comprises at least one component for, for each fast processed fast data packet processed on the faster path, determining whether one or more slow data packet precede the fast data packet in the original packet order; determining whether said one or more slow data packets have been processed; and, if one or more slow data packets that preceded said fast data packet in said original packet order has not been processed, assigning ~~in use assigns the~~ one or more fast processed

fast data packet that was processed by the faster processing unit to the memory ~~if not all slow data packets that before the processing were in order before the fast data packet was processed and given to the system output~~, and ~~wherein the second distributing unit in use gives~~ providing the fast processed fast data packets to the system output if all slow data packets that preceded the one or more fast data packets in the original packet order have been ~~before the processing were in order before the fast data packet was~~ processed and given to the system output~~, and wherein the second distributing unit in use gives the processed fast data packets to the system output after all in the memory stored fast data packets are drained out to the system output~~.

9. (currently amended) A data processing system as recited in claim 8, wherein the first distributing unit in use generates a sync signal if a slow data packet is followed by a fast data packet in the original packet order and assigns the sync signal to the slower processing unit, the slower processing unit in use generates a ready signal and gives the ready signal to the second distributing unit and the memory in response to processing of the sync signal, the second distributing unit in use puts the processed fast data packets in the memory until the ready signal is recognised, the memory in use provides ~~drains~~ the stored fast processed fast data packets to the system output after receiving the ready signal, and the second distributing unit in use assigns ~~the~~ successive fast processed fast data packets ~~after draining the stored fast data packets of the memory~~ to the system output directly.

10. (currently amended) A data processing system as recited in claim 8, comprising a packet counter connected to the input and the output of the slower processing unit

for detecting the number of slow data packets that are to be processed in the slower processing unit, for detecting the number of the processed slow data packets that leave the slower processing unit, and for giving a ready signal to the second distributing unit and the memory, if all counted slow data packets were processed, the memory draining the stored fast data packets to the system output on receipt of the ready signal, and the second distributing unit connecting the output of the faster ~~fast~~ path with the system output on drainage of the memory.

11. (currently amended) A data processing system as recited in claim 10, wherein the packet counter counts the slow data packets of different data flows, the system comprising a plurality of memories each for storing fast processed fast data packets of a separate data flow which have been processed faster ~~processed~~ in the faster processing unit than the slow data packets of the respective data flow that preceded the fast data packets in the original packet order for that data flow ~~prior to processing were in order before the fast data packets~~.

12. (currently amended) A computer readable medium containing computer executable instructions ~~program storage device readable by a digital processing apparatus and having a program of instructions which are tangibly embodied on the storage device and which are executable by the processing apparatus~~ to perform a method of altering a header of an incoming frame of network mode to a modified header of an outgoing frame, the method comprising:

depending on a predetermined feature of a data packet, said data packet being one of a plurality of data packets received in an original packet order, processing the data packet as one of a slow data packet on a slower path or a

fast data packet on a faster path, wherein the data packet is processed faster in the faster path than in the slower path; and

reordering the data packets ~~from the memory and outputting to the output when al the slow data packets that before the processing were in order before the fast data packets are received at the output.~~ after the processing into the original packet order;

wherein the reordering comprises the steps of:

for each fast processed fast data packet processed on the faster path, determining whether one or more slow data packet precede the fast data packet in the original packet order; determining whether said one or more slow data packets have been processed and received at an output; and, if one or more slow data packets that preceded said fast data packet in said original packet order has not been processed and received at an output, storing one or more fast processed ~~the~~ fast data packets that were processed on the faster path in a memory until said one or more ~~if not all the~~ slow data packets ~~that before the processing were in order before the fast data packets and were processed on the slower path are~~ have been received at an output; and,

fetching the stored one or more fast processed fast data packets from the memory for ~~and~~ outputting to the output when all the slow data packets that preceded the one or more fast data packets in the original packet order have been ~~before the processing were in order before the fast data packets are~~ received at the output.